Hybrid Machine Learning Algorithm for Predicting Distributed Denial of Service Attacks

Carine Umunyana, Dr. Kipruto W Cheruiyot , Dr. Michael Kimwele
Computing Department, Jomo Kenyatta University of Agriculture and Technology,
P.O.Box 62000-00200, Nairobi, Kenya
umucarin@gmail.com

Abstract: A distributed denial-of-service (DDoS) attack is one in which a multitude of compromised systems attack a single target. The flood of incoming messages to the target system essentially forces it to shut down. Two machine learning algorithms are used (Adaboost and Random forest) to detect DDoS attacks.

Keywords: Distributed denial of Service (DDoS) Attacks, Adaboost, Random forest, Hybrid machine learning algorithms

I. Introduction

Today, the number of attacks against large computer systems or networks is growing at a rapid speed. One of the major threats to cyber security is Distributed Denial-of-Service (DDoS) attack, in which the victim network element(s) are bombarded with high volume of fabricated attacking packets originated from a large number of Zombies. The aim of the attack is to overload the victim and render it incapable of performing normal transactions. To protect network servers, network routers and client hosts from becoming the handlers, Zombies and victims of distributed denial-of-service (DDoS) attacks machine learning approach can be adopted as a sure shot weapon to these attacks

Distributed Denial-of-Service (DDoS) attack is the one in which the victim‘s network elements are bombarded with high volume of fictitious attacking packets that originate from a large number of machines. A successful attack allows the attacker to gain access to the victim's machine, allowing stealing of sensitive internal data and possibly cause disruption and denial of service in some cases (Sonal R.Chakole, 2014).

Out of the various categories of DDoS attacks such as flooding, software exploit, protocol based etc Distributed Denial of service attack is the most famous. In fact, DDoS attack uses series of Zombies to initiate a flood attack against an unsafe single site. DDoS attack is initiated in 2-phases(Dongqi Wang, 2008) Recruiting phase and Action phase.

In Recruiting phase attacker initiates the attack from the master computer and tries to find some slave (Zombies) computers to be involved in the attack. A small piece of software is installed on the Zombies to run the attacker commands. The Action phase continued through a command issued from the attacker resides on the master computer toward the Zombies computers to run their pieces of software. The mission of the piece of software is to send dummy traffic designated toward the victim. The result is a massive flood of packets that crashes the host or swamp down the entire network operations. Very few networks or hosts can effectively cope with such a scale of attacks today. Most of the handler and Zombie are completely unaware of the fact that they were being used for launching of a DDoS attack (Rao, 2015).

"Learning is any process by which a system improves performance from experience"[Herbert Alexander Simon], Machine Learning is concerned with computer programs that can automatically adapt and customize themselves to individual users. Machine learning applications are computer software programs or packages that enable the extraction and identification of patterns from experience, this is categorized into four parties supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. In supervised learning the correct classes of the data are known. Unsupervised learning the correct classes of the training data are not known.

Semi-supervised learning is a mixed of supervised and unsupervised learning. Reinforcement learning allows the machine or software agent to learn its behaviour based on feedback from the environment (Wikipedia, 2015)

II. Material and Methodology

A Denial of Service (DoS) attack is an attempt by the attacker to prevent the legitimate users of a service from using that service. DDoS is a type of DOS attack where multiple compromised systems, which are often infected, are used to target a single system causing a Denial of Service (DoS) attack.

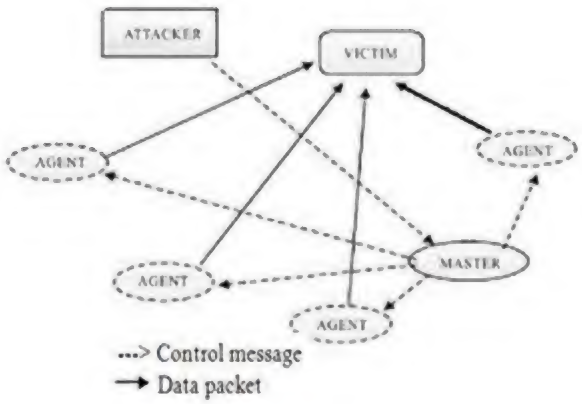


Figure 2.1 A DDoS structure

2.1 DDOS ATTACKS

DDoS attacks can be divided into five major categories that efficiently describe its architectural structure and overall behaviour. The first category, labelled level of computerization, specifies the attack‘s degree of automatization. The second category, named attack network, addresses the communication between the resources used for the actual attack and the source of the instruction initiating the event (zombies). Oppressed vulnerabilities are the next category for classifying a DDoS attack and describe the actual attack mechanism. The category influence characterizes a DDoS attack based on the attack‘s impact. The final category, attack intensity dynamics, consider the size of the attack related to the aspect of time (Usman Tariq, 2006).

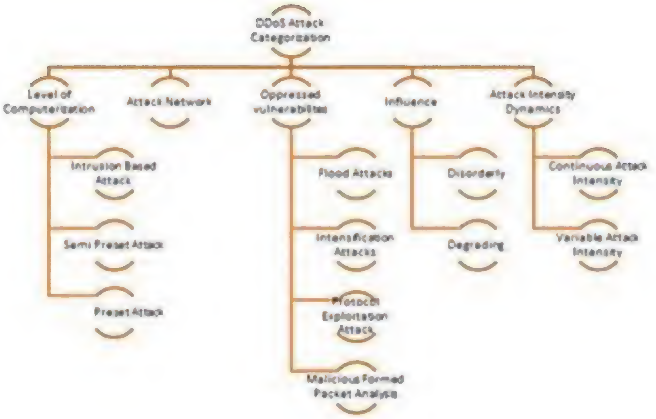

DDoS Attack Categorization
Level of Computerization
Attack Network
Oppressed vulnerabilities
Influence
Attack Intensity Dynamics
Intrusion Based Attack
Semi Preset Attack
Preset Attack
Flood Attacks
Intensification Attacks
Protocol Exploitation Attack
Malicious Formed Packet Analysis
Disorderly
Degrading
Continuous Attack Intensity
Variable Attack Intensity

Figure 2.2 A DDoS attack categorization

2.1.2 DDOS DEFENSES

The various DDoS defense mechanisms' characteristics can, as with the attacks, be structured into a similar scheme of categorizations. This scheme consists of four major categories, each subdivided into smaller fragments.

The first categorization is **submissive defence mechanism**, which initiates after the attack is detected. The second major categorization is **active defence mechanism**. This category is similar to the aforementioned. However, the significant difference is that active defence mechanisms are implemented in order to rapidly mitigate the attack by various measures.

Categorization by action is the next major characteristic that can be used to identify various DDoS defences. The characteristic defines the main purpose of the defence mechanism**.** The last major category is defence deployment position, which addresses the physical location of the **defence mechanisms' placement**. These characteristics address defence mechanism implemented close to the source of the attack.

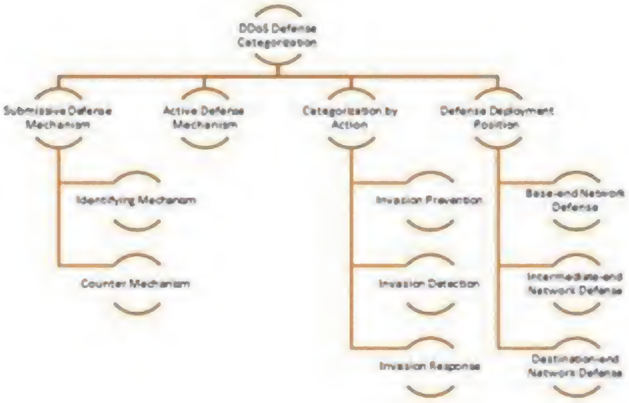
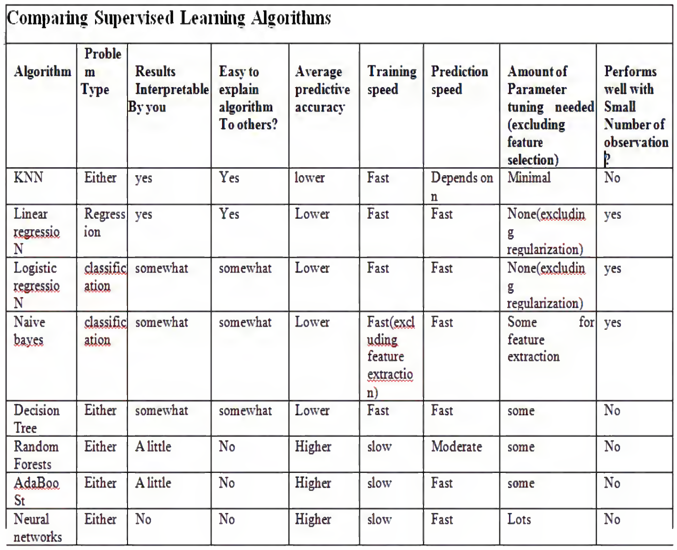


Figure 2.3 A DDoS defense mechanism

2.2 MACHINE LEARNING

A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.



Comparing Supervised Learning Algorithms								
Algorithm	Problem Type	Results Interpretable By you	Easy to explain algorithm To others?	Average predictive accuracy	Training speed	Prediction speed	Amount of Parameter tuning needed (excluding feature selection)	Performs well with Small Number of observation?
KNN	Either	yes	Yes	lower	Fast	Depends on n	Minimal	No
Linear regressioN	Regression	yes	Yes	Lower	Fast	Fast	None(excluding regularization)	yes
Logistic regressioN	classification	somewhat	somewhat	Lower	Fast	Fast	None(excluding regularization)	yes
Naive bayes	classification	somewhat	somewhat	Lower	Fast(excluding feature extraction)	Fast	Some for feature extraction	yes
Decision Tree	Either	somewhat	somewhat	Lower	Fast	Fast	some	No
Random Forests	Either	A little	No	Higher	slow	Moderate	some	No
AdaBooSt	Either	A little	No	Higher	slow	Fast	some	No
Neural networks	Either	No	No	Higher	slow	Fast	Lots	No

Table 2.1 Supervised learning algorithm classification

2.3.3 Hybrid Adaboost and Random Forests

For the combination of AdaBoost and random forests (ABRF) technique used, we utilized the random forest as a weak learner to generate the prediction models with less error rate. Although AdaBoost works fast with simple weak learners, random forest is of interest in our real world data set, due to few research studies having employed this method to predict in the networking domain. Thirteen steps of the hybrid AdaBoost and random forests algorithm.

```
Input: S: training set, S=$x_i$(i=1,2,...,n), labels $y_i \in Y$
        K: Iterations number
        L:Learn (Random Forests algorithm as weak learner)
        f: number of input instance to be used at each of the tree
        B: number of generated trees in random forest
1) Assign N sample $(x_1,y_1),..,(x_n,y_n)$; $xi \in X$, $y_i \in \{-1,+1\}$
2) Initialize the weights of $D_1(i)=1/n$, i=1,...,n)
3) for k=1,...,K
4) empty E with the distribution $D_k$
5) for b=1to B
6) $S_b$ = boostrapSample(S)
7) $C_b$ = BuildRandomTreeClassifiers($S_b$,f)
8) $E=E \cup \{C_b\}$
9) next b
10) Get weak hypothesis $h_k$:X{-1,+1} with its error: $\varepsilon_k = \sum_{i=h_k(x_i)\neq y_i} D_k(i)$
11) Update distribution $D_k : D_{k+1}(i) = \frac{D_k(i)\exp(-\alpha_k . y_k h_k(x_k))}{z_k}$
12) next k
13) Output : $H(x) = sign\left(\sum_{k=1}^{k} \alpha_k h_k(x)\right)$
```

Table 2.4 Hybrid AdaBoost and Random Forests

This combination has advantages including increased performance and prediction ability of the models in some data sets. The results obtained that the combination has a low error rate. Error rate is the basic measurement method, which is used to investigate the weak and strong points of algorithms.

III. Results and Tables

The simulations were done using the Packet Tracer simulation and Wireshark 2.0.2. The results presented for each value are the average of 6 simulation runs and simulation parameters took the following values:

The principal metric in tests is the percentage of detections, which is assessed in terms of misbehaviour threshold.

In the research, network traffic simulation is represented on the figure 4.1 shown

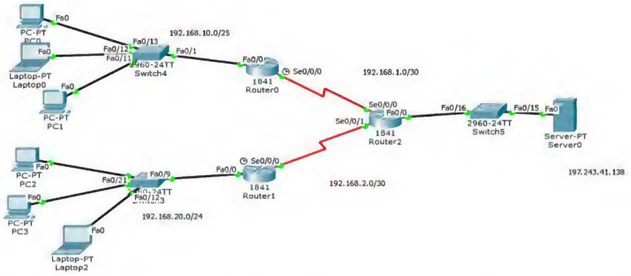


Figure 4.1, Normal traffic of the packets without anomalies

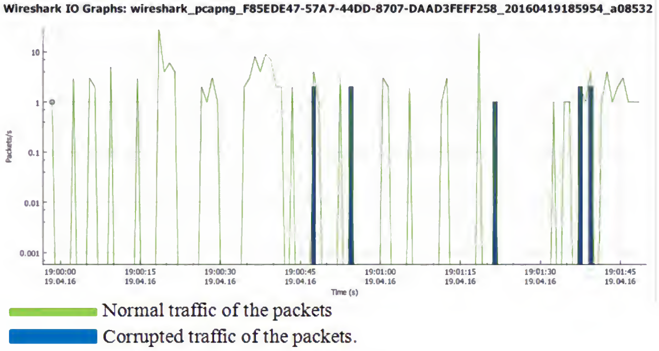


Figure 4.2 Wireshark traffic analysis of the packets.

4.2 Data collected

The following table represents the data obtained from IPRC East (Technical College) in Rwanda; in April 2016.

	T1	T2	T3	T4	T5	T6
Not corrupted	95,84%	95.35%	87.2%	86,55%	88,5%	95.59%
Corrupted	4.16%	4.65%	12.8%	13.45%	11.5%	4.41%

Table 4.1 data collected using wireshark

After analyzing these data, the average of the data above is 91.5% of the normal traffic (not corrupted packets) and 8.5% of the abnormal (corrupted packets), comparing these analyzed data with the report (Akamai, Q2 2015) that says 92, 2% is normal traffic (not corrupted packets) and 7.7% is abnormal traffics (corrupted packets) from the other country where Africa is located and that have been detailed in the report.

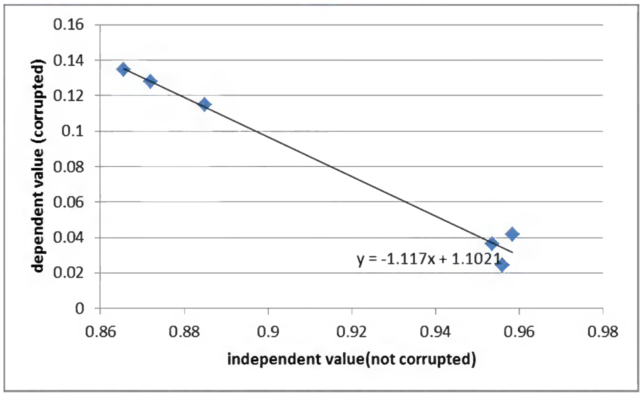
4.3 Adaboost Algorithm implementation

In these experiments, the performance and effectiveness of the proposed algorithm is done with six data sets, and the output of the experiment shows that, as the traffic of not corrupted packet are increasing the corrupted packets are decreased, presented in Figure 4.3.

Y: dependent values (corrupted)

X: independent values (not corrupted)

$\hat{\beta}_1$ =-1.11698 $\hat{\beta}_0$ =1.102062



4.4 Random Forests implementation

In these experiments, the performance and effectiveness of the proposed algorithm is done with six data sets, and the output of the experiment shows that, each traffic packets is independent and it is contains corrupted data and not corrupted data , as shown in Figure 4.4.

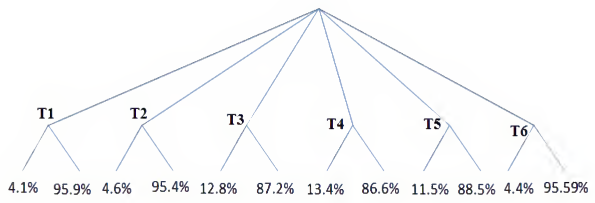
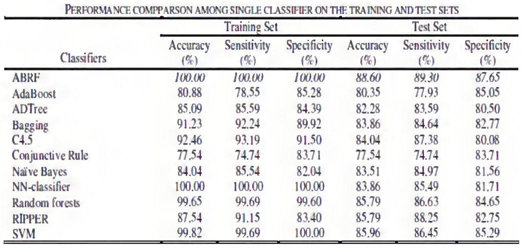


Figure 4.4 Relationship between corrupted and not corrupted data using Random forest.

4.5 Hybrid Adaboost and Random forest

In these experiments, the performance and effectiveness of the proposed algorithm is compared with 10 single classifiers (Guandong Xu, 2008).

TABLE I
PERFORMANCE COMPPARSON AMONG SINGLE CLASSIFIER ON THE TRAINING AND TEST SETS



Classifiers	Training Set			Test Set		
	Accuracy (%)	Sensitivity (%)	Specificity (%)	Accuracy (%)	Sensitivity (%)	Specificity (%)
ABRF	*100.00*	*100.00*	*100.00*	*88.60*	*89.30*	*87.65*
AdaBoost	80.88	78.55	85.28	80.35	77.93	85.05
ADTree	85.09	85.59	84.39	82.28	83.59	80.50
Bagging	91.23	92.24	89.92	83.86	84.64	82.77
C4.5	92.46	93.19	91.50	84.04	87.38	80.08
Conjunctive Rule	77.54	74.74	83.71	77.54	74.74	83.71
Naïve Bayes	84.04	85.54	82.04	83.51	84.97	81.56
NN-classifier	100.00	100.00	100.00	83.86	85.49	81.71
Random forests	99.65	99.69	99.60	85.79	86.63	84.65
RIPPER	87.54	91.15	83.40	85.79	88.25	82.75
SVM	99.82	99.69	100.00	85.96	86.45	85.29

4.5.1Model Selection

In these experiments, the performance of the proposed algorithm (ABRF) is compared with three classifiers including AdaBoost, random forests and C4.5 using the ROC curve. The experiment results were given in Figure 4.5.

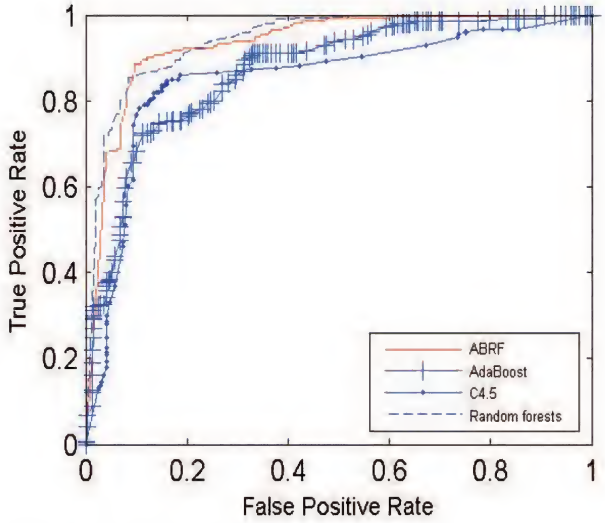


Figure 4.5 ROC curve

Figure 4.5 illustrates the predictive performance of four classifiers including AdaBoost, random forests, ABRF and C4.5. The results show that ABRF algorithm improves the prediction ability of random forests in some points and performs relatively well compared with AdaBoost and C4.5 in terms of ROC curve. However, it is hardly possible to distinguish the difference in performance between ABRF and random forests models in ROC curve. Therefore, the advance techniques used to select these models such as AUC scores is needed. The experiment results were shown in Figure 4.6.

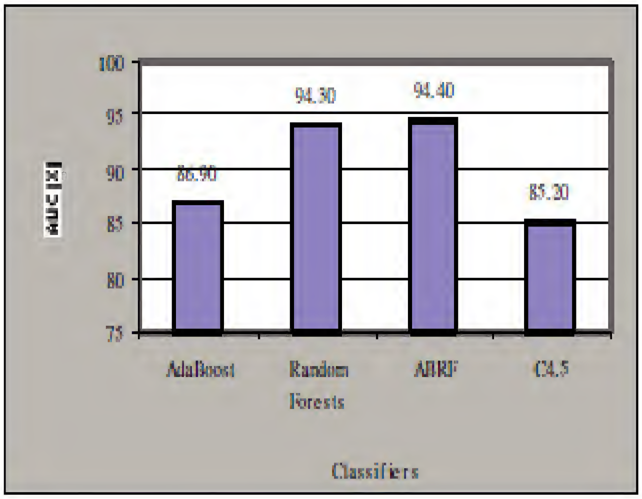


Figure 4.6 AUC scores

IV. Conclusion

In this paper we proposed a combination of the AdaBoost and random forests algorithms for constructing distributed denial of service prediction algorithm. We illustrated the capability and effectiveness of the hybrid machine learning algorithms (adaboost and random forest). Finally, a prediction using hybrid machine learning algorithms would be of interest.

Acknowledgement

I thank God the Almighty who has provided me the breath of life to enable the successful completion of this work and for His unconditional love. I express my heart-felt appreciation to my late parents who have provided me all that is required to allow me reaching this academic level.

References

i. *Agarwal, S. (2005). Generalization Bounds for the Area Under the ROC Curve. Journal of Machine Learning Research, 393–425 .*

ii. *Akamai. (Q2 2015). Global DDoS Threat Landscape . Incapsula.*

iii. *Alireza Shameli Sendi, M. D. (2012). Real Time Intrusion Prediction based on Optimized Alerts with Hidden Markov Model. JOURNAL OF NETWORKS.*

iv. *Alonso, J. (2011). Predicting Software Anomalies Using Machine Learning Techniques. 10th IEEE International Symposium (pp. 163 - 170). Cambridge: IEEE.*

v. *Antonio Colella, C. M. (2014). Amplification DDoS Attacks: Emerging Threats and Defense Strategies. IFIP* *vi. International Federation for Information Processing, 298–310.*

vii. Ben-David, S. S.-S. (2014). Understanding Machine Learning:From Theory to Algorithms. Cambridge University Press.

viii. Bishop, C. M. (1995). Neural Networks for Pattern Recognition. New York, NY, USA: Oxford University Press.

ix. Breiman, L. (2001). Random Forests. Journal of Machine Learning vol.45, 5-32.

x. Cambridge. (2015). Akamai Releases Second Quarter 2015 'State Of The Internet' Report. Akamai.

xi. Christos Douligeris, A. M. (2003). DDoS attacks and defense mechanisms: classification and state-of-the-art. Elsevier.

xii. Dongqi Wang, G. C. (2008). Research on the Detection of Distributed Denial of Service Attacks Based on the Characteristics of IP Flow. IFIP International Federation for Information Processing , 86–93.

xiii. G. Pack, J. Y. (2006). On Filtering of DDoS Attacks Based on Source Address Prefixes. in Proceedings of the 2nd International Conference on Security and Privacy in Communication Networks (SecureComm 2006).

xiv. G.Florance. (2015). Survey of IP Traceback Methods in Distributed Denial of Service Attacks. International Journal of Innovative Research in Science, Engineering and Technology.

xv. Guandong Xu, Y. Z. (2008). AdaBoost Algorithm with Random Forests for Predicting Breast Cancer Survivability. International Joint Conference on Neural Networks.

xvi. Guoxing Zhang, S. J. (2009). A Prediction-based Detection Algorithm against Distributed Denial-of-Service Attacks. ResearchGate. Shengming Jiang.

xvii. Guy Leshem, a. Y. (2007). Traffic Flow Prediction using Adaboost Algorithm with Random Forests as a Weak Learner. International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering.

xviii. Harvard, A. S. (1988, November 3). The What, Why, and How of the 1988 Internet Worm. Retrieved January 3, 2016, from snowplow.org: https://snowplow.org/tom/worm/worm.html

xix. Hastie. (2002, june 5). Trevor_Hastie. Retrieved January 12, 2016, from wikipedia.org: https://en.wikipedia.org/wiki/Trevor_Hastie

xx. Jackman. (2007). Prediction with Linear Regression Models. Departement of Political Science Standford University.

xxi. Jaree Thongkam, G. X. (2008). Breast Cancer via Adabooost Algorithms. school of Computer Science and Mathematics Victoria University.

xxii. Jelena Mirkovic, M. I. (2013). D-WARD: A Source-End Defense Against Flooding Denial-of-Service Attacks.

xxiii. Jin Huang, C. X. (2005). Using AUC and accuracy in evaluating learning algorithms. IEEE Transactions on Knowledge and Data Engineering.

xxiv. Kamber, J. H. (2006). Data Mining:Concepts and Techniques. Elsevier.

xxv. Kanwal Garg, R. C. (2011). DETECTION OF DDOS ATTACKS USING DATA MINING . International Journal of Computing and Business Research (IJCBR).